PetaScale Calculations of the Electronic Structures of Nanostructures with Hundreds of Thousands of Processors

Lin-Wang Wang, Zhengji Zhao and Juan Meza

OVERVIEW:

Density functional theory (DFT) is the most widely used *ab initio* method in material simulations. It accounts for 75% of the NERSC allocation time in the material science category. The DFT can be used to calculate the electronic structure, the charge density, the total energy and the atomic forces of a material system. With the advance of the HPC power and new algorithms, DFT can now be used to study thousand atom systems in some limited ways (e.g, a single selfconsistent calculation without atomic relaxation). But there are many problems which either requires much larger systems (e.g, >100,000 atoms), or many total energy calculation steps (e.g. for molecular dynamics or atomic relaxations). Examples include: grain boundary, dislocation energies and atomic structures, impurity transport and clustering in semiconductors, nanostructure growth, electronic structures of nanostructures and their internal electric fields. Due to the $O(N^3)$ scaling of the conventional DFT algorithms (as implemented in codes like Qbox, Paratec, Petots), these problems are beyond the reach even for petascale computers. As the proposed petascale computers might have millions of processors, new computational paradigms and algorithms are needed to solve the above large scale problems. In particular, O(N) scaling algorithms with parallelization capability up to millions of processors are needed. For a large material science problem, a natural approach to achieve this goal is by divide-and-conquer method: to spatially divide the system into many small pieces, and solve each piece by a small local group of processors. This solves the O(N) scaling and the parallelization problem at the same time. However, the challenge of this approach is for how to divide the system into small pieces and how to patch them up without the trace of the spatial division. Here, we present a linear scaling 3 dimensional fragment (LS3DF) method which uses a novel division-patching scheme that cancels out the artificial boundary effects of the spatial division. As a result, the LS3DF results are essential the same as the original full system DFT results (with the difference smaller than chemical accuracy and smaller than other numerical uncertainties, e.g, due to numerical grids), while with a required floating point operation thousands of times smaller, and computational time thousands of times shorter, than the conventional DFT method. For example, using a few thousand processors, the LS3DF can calculate a >10,000 atom system within an hour while the conventional method might take more than a month to finish. The LS3DF method is applicable to insulator and semiconductor systems, it covers a current gap in DOE's material science code portfolio for *ab initio* ultrascale simulation. We will use it here to solve the internal electric field problems for composite nanostructures.

SCIENTIFIC PROBLEM:

Nanostructures such as quantum dots and wires, composite quantum rods and core/shell structures have been proposed to be used as electronic devices or optical devices like solar cells. To understand the electronic structures of such systems and the corresponding carrier dynamics is essential to the successful designs and deployments of such devices. Despite of more than a decade of research, some critical issues of the electronic structure of moderately complex nanostructures are still poorly understood. One such issue is the internal electric field in a composite colloidal nanostructure and its consequences on the electron wavefunctions. It is well known that there are strong internal electric fields in some of the bulk semiconductor heterostructures, like the InN/GaN superlattice. These electric fields could be caused by surface and interface dipoles, total dipole of the nanostructure, piezoelectric effects, surface trapped charges and charged dopants. They induce strong spatial localizations of the wavefunctions, thus cause different electron-hole recombination rates, charge transports, and nonlinear optical properties, all important to the performance of the nanostructure electro-optical devices. Unfortunately, the continuous model used in conventional device simulations can no

longer be used for these nanostructures due to the atomic natures of the charge, dopant and geometry, the high order effects of different phenomenon, and the change of dielectric functions, etc. Thus, what needed here is a direct atomistic *ab initio* selfconsistent calculation for the charge density and the electric field, and the corresponding atomic relaxation for the nanosystem. Since the atomic number N of such composite nanostructures easily exceeds 10,000 or even 100,000 atoms, the traditional $O(N^3)$ scaling DFT method cannot be used. Partly because of this, the internal electric field problem remains to be one of the most outstanding unsolved problems in colloidal nanoscience. For example, we don't even know whether there is a large internal electric field in a simple quantum dot consisted with dipolar semiconductors.

Here, we propose to use our newly developed LS3DF method to calculate the ~10,000–100,000 atom composite nanosystems using the local density approximation (LDA) of the DFT. We will investigate nanostructures with different geometries and heterostructure composites: e.g, A/B nanorod, A/B core/shell nanowire, A/B/A dumbbell structures. We will study the effect of different surface passivation and surface termination layers, e.g, the cation ended (0001) bottom layer in a wurtzite nanostructure or anion ended bottom layer. We will study the effect of a surface charge, e.g, a charge trapped in a surface dangling bond. Lastly, we will study the effect of a single dopant in a nanostructure, and test the bulk concepts of p-type and n-type semiconductors, and p-n conjunctions in colloidal nanosystems. Our theoretical calculation can help the experimentalists to design better solar cell using nanostructures, which has been identified as one of the possible interruptive technologies to revolutionize the solar cell field, hence to solve the world energy problem with zero carbon dioxide emission.

ALGORITHM:

With the increase number of processors in large supercomputers, and the push to petascale computation, the ability to massive parallelization is an essential issue. While planewave LDA codes like the Qbox has demonstrated its capability of using hundreds of thousands of processors on the BlueGene/L computer[1], the intrinsic $O(N^3)$ scaling for its floating point operation makes it not necessarily the most efficient way to solve a given science problem, like the one posted above. In a way the $O(N^3)$ floating point operation dwarf the $O(N^2)$ (the wavefunction memory) communication needed, thus mitigate the massive parallelization problem. But to solve the same problem faster, one does need a O(N) scheme.

In the past 10 years, there has been a great deal of research concerning O(N) *ab initio* methods [1]. Most of these methods use localized orbitals, and minimize the total energy as a function of these localized orbitals. Unfortunately the use of localized orbitals can introduce local minima in the total energy functional, which makes the total energy minimization difficult. Additionally, there is no O(N) code that can effectively use thousands of processors because the strong overlaps between the local orbitals make parallelization a nontrivial task. As a result of these challenges, in spite of a decade of intense research, the application of current O(N) methods is still quite limited. Another O(N) approach, the LSMS method [2] scales to thousands of processors, but this method can only be applied to metals, not the semiconductor nanostructures. It is often used to study metallic alloys and magnetic systems.

Our LS3DF is based on the observation that the total energy of a system can be decomposed into the quantum mechanical part (the wavefunction kinetic energy and the exchange correlation energy), and the classical electrostatic part. While the electrostatic energy (Coulomb energy) is long range, the quantum mechanical energy is local in nature (short sighted). While the long range Coulomb interaction can be solved efficiently by the Poisson equation even for million-atom systems, the quantum mechanical part is the most difficult one to be solved. For this, we will take advantage of its locality by using the aforementioned spatial decomposition divide-and-conquer method. While there are previous methods [3, 4] based on this divide-and-conquer concept, they all rely on positive spatial partition functions to divide and patch the spaces. There are intrinsic difficulties to use this positive partition function technique, especially for dividing the kinetic

energies. In contrast, our novel division-patching method avoids these problems, thus result in a much more accurate algorithm (with the accuracy essential the same as the original full system LDA method).

Our LS3DF spatial division-patching technique is inspired by the fragment molecular orbital (FMO) method proposed by Kitaura *et al* [5,6] and combined with the ideas from our own charge patching method [7]. FMO is used for organic chain like molecules. In FMO, the long chain molecule is chopped into fragment pieces. Then each piece and pairs of nearby pieces are calculated. The electron charge density is then added up from all the pieces and their pairs, with positive sign for the pair and negative sign for the pieces itself. The usage of pairs and negative signs are innovative as this allows the calculation of the energy of the artificial boundaries, which can subsequently be subtracted from the total energy and charge density summation. Our LS3DF method extend this technique to arbitrary 3 dimensional system. Instead of using pairs of pieces, we divide the system using overlapping regions (pieces, fragments). More specifically, our division scheme is illustrated in Fig.1 for a 2 dimensional system for clarity. Here, a supercell is divided into m1x m2 grid points. From each grid point corner (i1,i2), we can defined four pieces with dimension: 1x1, 1x2, 2x1, 2x2. Note that, they are overlapping pieces. Now, after all the pieces at all the (i1,i2) corners are calculated, the total charge density of the whole system can be patched together as: $\rho_{tot}(r) = \sum_{(i1,i2),D} sign_D \rho_{(i1,i2),D}(r)$, here D denotes the dimension 1x1, 1x2, 2x1, 2x2, and the $sign_D$ is + for 1x1 and 2x2, and – for 1x2, 2x1. The total energy (especially the kinetic energy part) can be expressed in similar fashion using the wavefunctions of each pieces, although the electron-electron Coulomb interaction is expressed based on the total charge density _tot(r). To make sense out of the above formula, we can check each point inside a piece (A point in Fig.1). Note that, each spatial point will be included in 3^2 pieces: 4 "2x2" pieces, 2 "2x1" pieces, 2 "1x2" pieces, and 1 "1x1" piece. After the above +/- cancellations, it will be covered by only one piece, that is what we need. We can also check for each boundary point. A boundary can be defined with a direction (i.e, boundary from A to B is different than boundary from B to A, we has used an arrow in Fig.1 to represent a directional boundary). A given directional boundary is covered by 6 pieces, with equal numbers of positive and negative signs. Since all these pieces have the same (directional) boundary at that point, and given the short sight-ness, their charge density will be the same near that point. As a result, the boundary effect will be cancelled out. The same is true for the corner effects. This division scheme can be extended to 3 dimension, where at each corner point (i1,i2,i3), there will be eight pieces: 1x1x1, 1x1x2, 1x2x1, 2x1x1, 1x2x2, 2x1x2, 2x2x1, 2x2x2. In this case, each spatial point will be covered by 3^3 pieces. The sign in the formula is positive for 2x2x2, 1x1x2, 1x2x1, 2x1x1, while negative for 2x2x1, 2x1x2, 1x2x2, 1x1x1.

The short sight-ness comes from an energy gap between the occupied states and the unoccupied states in the semiconductor system. In order to keep this short sight-ness in each piece, we need to maintain an energy gap in each piece. This is achieved by a proper surface passivation (usually with the H atoms) for the dangling bonds in the artificially created boundaries. Each small piece with a small vacuum margin is solved using the conventional planewave codes (PEtot [7]) with a small number of processors (e.g., 16). This allows us to take the full advantages of the planewave code (e.g, the better analytical properties over the real space code). The Fast Fourier Transform (FFT) is done within these 16 local processors. In the future, each piece can be solved by one chip with its multiple cores. When the wavefunctions are solved for each piece, there is no communication need between the pieces (hence between the 16 processor groups). Since the accuracy of this method depends only on the size of the pieces, for larger systems, more pieces (with similar sizes as before) will be generated. This makes the total floating point operation scales as O(N) and also makes it naturally parallelizable to very large

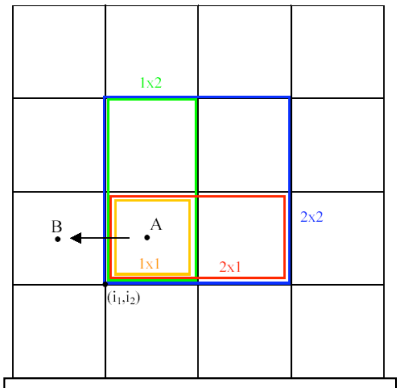

Fig. 1, The division of space and fragment pieces from corner (i_1,i_2).

number of processors (as each 16 processor group calculates a few pieces independently). Thus we believe the LS3DF method is a very good candidate for petascale calculations.

The LS3DF result can be extremely accurate when compared with the original full system direct LDA result. For example, if we take the cubic 8 atom cell in a diamond Si structure as our smallest 1x1x1 piece, and use this method to calculate a Si box passivated with H atoms, we found that the relative energy error is 8.E-6, smaller than the typical error introduced by other sources of the numerical approximations (e.g., the numerical basis set truncation). The absolute energy error is less than 4 meV/atom=0.1Kcal/mol, thus better than typical chemical accuracy. The total electron charge has a relative error of 0.03%, thus essentially the same as the direct calculated results. The atomic force error is about 6.4E-5 a.u, which is an order of magnitude smaller than typical stopping criterion used in *ab initio* atomic relaxation. Thus, for practical purposes, the result of the 3DF method is essential the same as in a direct full LDA calculation.

CODE AND SCALING:

The LS3DF code is based on the planewave DFT PEtot code [8]. The flow chart of the LS3DF code is shown in Fig.2 (right side), in comparison to the original LDA code (left side). The LS3DF code consists of several components: PEtot_F, which divides the number of processors into 16-processor groups, and calculates the fragment eigen wavefunctions $\psi_{F,I}$ by each group for a given fragment potential $V_F(r)$ using conjugate gradient method. It also calculates the fragment charge density $\rho_F(r)$ from the wavefunction $\psi_{F,I}$; Gen_dens patches together the fragment charge densities $\rho_F(r)$ to generate the total charge density $\rho_{tot}(r)$ of the whole system. The Poisson step generates the LDA total potential $V_{tot}(r)$ from the total charge density $\rho_{tot}(r)$. This step solves the Poisson equation for the whole system using a global FFT. It also uses the Pulay scheme to mix the resulting LDA potential that is used in the next iteration. Finally, Gen_VF generates the fragment potential $V_F(r)$ from the input total potential $V_{tot}(r)$.

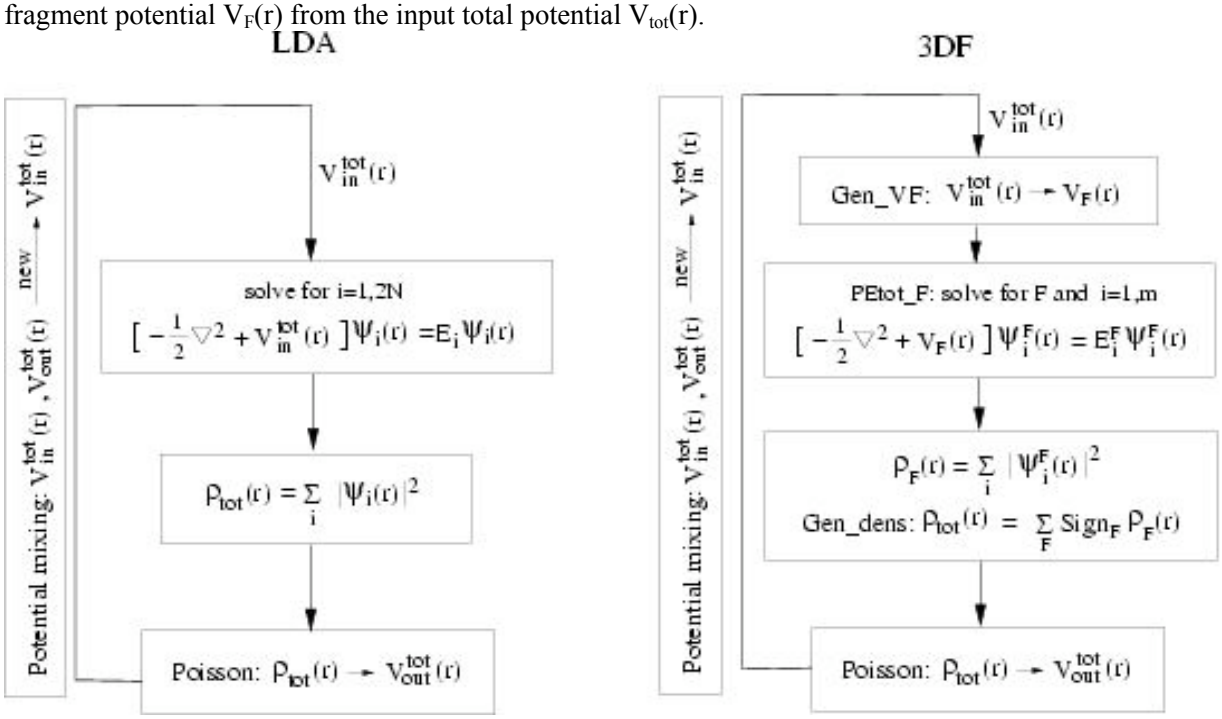

Fig. 2. Program flow chart for a conventional LDA method and the LS3DF method

All the codes in Fig. 2 are parallelized. As a demonstration, we have used this approach to calculate a ~3000 atom Si quantum dot passivated by H atoms (more exactly $Si_{2253}H_{652}$). The calculated total charge density of this quantum dot is shown in Fig.3 a. The calculation of this system took approximately 2 hours on 1024 processors of the NERSC seaborg computer.

In Fig. 2, the PEtot_F step is the most time consuming part. The scaling of this part for the 3000 quantum dot (QD) test case is shown in Fig. 3 b on the seaborg computer. As can be seen, this step scales well up to 1024 processors. We believe it should scale effectively to tens of thousands of processors since each small group (16 processors) solves the fragments independently in PEtot_F.

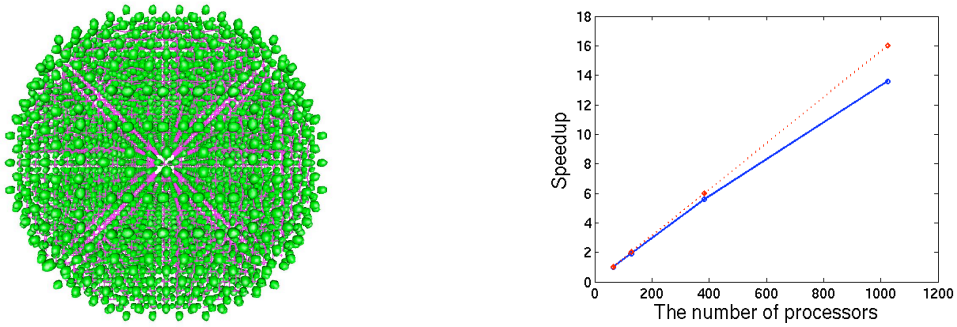

Fig. 3. a) the charge density isosurface (green) plot of a ~3000 atom Si quantum dot passivated by H atoms. The pink color indicates the bonds between Si atoms. b) the speedup as a function of the number of processors used in PEtot_F. The red dashed line is the ideal scaling line.

Note that there are about 4605 fragments in this test case. Thus, the number of fragments is slightly larger than the total number of atoms. The number of atoms in each fragment ranges from 2 to 150 atoms. By using 1024 processors, we have 64 groups (assuming 16 processors/group). Thus each group will solve on average 72 fragments. If we assign each group to solve only one fragment, we could use 4605x16=73680 processors for the ~3000 atom QD problem. This could cause a serious load imbalance problem however. To avoid this problem, we propose reducing the number of groups to about half this, or roughly about 40,000 processors. We believe that the number of processors that we can efficiently use will be on the order of 10 times the number of atoms in the system. Thus for a 100,000 atom system, we should be able to efficiently use a million processors. For our 3,000 Si atom QD, when we used 1024 processors, the PEtot_F part of each self-consistent iteration took about 10 minutes. The Poisson solver we used in this calculation was based on FFTs. For the 3,000 atom Si QD case, the real space numerical grid employed was 240_240_240 and it took about 1 minute to solve the Poission equation using 128 processors. It took about half a minute each to finish the Gen_dens and Gen_VF program using 128 processors. Thus, in total, for the 1024 processors calculation above, it took approximately 12 minutes to finish one self-consistent iteration. For such a large QD, it typically takes 10 to 20 steps (the outer loop in Fig. 2) to converge the self-consistent iteration. As a result, it will take about 2 to 4 hours to finish one self-consistent calculation (for a fixed atomic position) for this type of QD system using 1024 processors.

REFERENCES:

[1] F. Gygi, *et al.,* Proceedings of Supercomputing, 2005 (ACM,2005)
[1] G. Goedecker, Rev. Mod. Phys., 7**1**, 1085 (1999).
[2] http://www.psc.edu/general/software/packages/lsms/
[3] W. Yang, Phys. Rev. Lett. 66, 1438 (1991).
[4] F. Shimojo, R.K. Kalia, A. Nakano, P. Vashishta, Comp. Phys. Commun. 167, 151 (2005).
[5] K. Kitaura, E. Ikeo, T. Asada, T. Nakano, and M. Uebayasi, Chem. Phys. Lett., **313**, 701(1999).
[6] K. Kitaura, S.-I. Sugiki, T. Nakano, Y. Komeiji, and M. Uebayasi, Chem. Phy. Lett, **336**, 163 (2001).
[7] L.W. Wang, J. Li, Phys. Rev. B 69, 153302 (2004).
[8] http://hpcrd.lbl.gov/~linwang/PEtot/PEtot.html